

# Program #4 - Connect4 plus plus

*Connect4. Plus more.*

Logistics:

- Due: **Fri Jun 6, 2014**
- Worth: 10 points, or 10% of your grade

We'll build our own linked lists and use them in this console game.

My source for this program is a UIC programming course:

[sites.google.com/site/uiccs102/program-3-connect](http://sites.google.com/site/uiccs102/program-3-connect)

## 1. Description

Have you ever played Connect4? If not... [www.coolmath-games.com/0-connectfour/](http://www.coolmath-games.com/0-connectfour/)

Connect4 plus plus (C4++) adds a twist to regular old Connect4, allowing you to remove pieces from the bottom. The goal is the same: to get 4 of your pieces in a row... to win! Our version will be a console game. There's a sample session at the end of this document that makes things clearer.

Some additional rules and error cases:

- Moves should be numbered (as in the sample session)
- If removing a piece results in a win for both players, then the win goes to the player whose turn it is
- Invalid move - adding a piece to a full column
- Invalid move - removing a piece from an empty column
- Invalid move - removing a piece that's not yours

## 2. Design

Try a design on paper.

- Identify the classes in this problem... be as literal as you can
- Specify as many methods for each class as you can
- Write any pseudo-code you can for these methods

Flex your design - Could your design be used in a GUI implementation?

Flex #2 - Could your design be used with a computer player?

We'll review and share designs in class and then go!

Code in one manageable chunk at a time. Here are some possible coding steps:

1. Print the welcome
2. Print the empty board
3. Add pieces to the top
4. Add command loop
5. Remove pieces from the bottom
6. Add check for Connect4 (along with a game winner!)

Step 6, the Connect4 check is a toughie. Do it piece-by-piece: 1) vertical Connect4, 2) horizontal Connect4, 3) diagonal Connect4.

**One special requirement** - Please create your own linked list class to keep track of: the columns in a game board and the pieces in a column  
I recommend using `LinkedList` to start out. Insert your own list at the end.

### 3. Grading

Please place the following in your `program4` folder on the k: drive:

- A `README` file describing the state of your program
- Your beautiful Java code that follows our Java Coding Guidelines
- Write your own linked list class for this program, rather than using the JCF

Good luck!



## Appendix A.

Here's a sample session... user input in **bold**:

```
****
****
    -> Program 4 - Connect 4 plus plus
****
****
```

```
Enter name of player 1(X) -> Bill
Good luck, Bill
```

```
Enter name of player 2(O) -> Holly
Good luck, Holly
```

```
-----
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
-----
 1 2 3 4 5 6 7 <- add piece
  A B C D E F G <- remove piece
```

1. Bill (X): Enter column for move -> **4**

```
-----
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |X| | | |
-----
 1 2 3 4 5 6 7 <- add piece
  A B C D E F G <- remove piece
```

2. Holly (O): Enter column for move -> **4**

```
-----
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |O| | | |
| | | |X| | | |
-----
```

```
1 2 3 4 5 6 7 <- add piece
A B C D E F G <- remove piece
```

3. Bill (X): Enter column for move -> **3**

```
-----
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |O| | | |
| | |X|X| | | |
-----
```

```
1 2 3 4 5 6 7 <- add piece
A B C D E F G <- remove piece
```

4. Holly (O): Enter column for move -> **5**

. . . game continues . . .

```
-----
| | | | | | | |
| | | | | | | |
| | | |O| | | |
| | | |X| | | |
|X|X|X|O|O| | |
|O|X|X|X|O|O|O|
-----
```

```
1 2 3 4 5 6 7 <- add piece
A B C D E F G <- remove piece
```

15. Bill (X): Enter column for move -> **A**  
Sorry. You can only remove your own piece.  
Please try again.

15. Bill (X): Enter column for move -> **D**

```
-----
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |O| | | |
|X|X|X|X|O| | |
|O|X|X|O|O|O|O|
-----
```

```
1 2 3 4 5 6 7 <- add piece
A B C D E F G <- remove piece
```

Connect4!  
Congrats Bill - you are a winner!  
Program #4 done.